Problems Solved:

16	17	18	19	20

Name:

Matrikel-Nr.:

TMCheck1k01k1

Problem 16. Construct a Turing machine $M = (Q, \Gamma, \sqcup, \{0,1\}, \delta, q_0, F)$ such that $L(M) = \{1^k 0 1^{k+1} \mid k \in \mathbb{N}\}$. Write down Q, Γ, F and δ explicitly.

Solution of Problem 16:

Informally we have:
If M is in a state q_0 and reads the sequence $0, 1, \sqcup$, then goes into an accepting state.

If the sequence starts with $\sqcup$ then goes into nonaccepting state.

If the first symbol is a 1, it shall be overwritten by a $\sqcup$ and then go to the next $\sqcup$. If the symbol before that next $\sqcup$ is a 1, overwritte it by a $\sqcup$, otherwise, i.e., the symbol before the $\sqcup$ is a 0, then M goes into nonaccepting state and terminates.

Then the head goes left to the first non-$\sqcup$ symbol and M goes into q_0 state.

The above procedure repeats until M terminates.

Formally we have:
$Q = \{q_0, q_1, q_2, q_3, q_4, q_5, q_6\}$, $\Gamma = \{0, 1, \sqcup\}$, $F = \{q_3\}$ and δ defined as follows:

δ	0	1	$\sqcup$	
q_0	$q_1 0 R$	$q_4 \sqcup R$	$-$	check for $01\sqcup$; overwritte 1 by $\sqcup$
q_1	$-$	$q_2 1 R$	$-$	check for $1\sqcup$
q_2	$-$	$-$	$q_3 \sqcup R$	check for $\sqcup$
q_3	$-$	$-$	$-$	accepting state
q_4	$q_4 0 R$	$q_4 1 R$	$q_5 \sqcup L$	go right to the end
q_5	$-$	$q_6 \sqcup L$	$-$	remove the last 1
q_6	$q_6 0 L$	$q_6 1 L$	$q_0 \sqcup R$	go left to the beginning

TM-LCM-En

Problem 17. Give an informal description of a Turing machine M which for any two natural numbers computes their least common multiple. You may assume that the numbers are represented in their unary form, e.g., the pair of numbers 5 and 3 is represented as 111110111.
Hint: You may use a Turing machine with two tapes, as described in the lecture notes.

Solution of Problem 17:

Write the first number on tape 1 and the second on tape 2. Then check whether both tapes have the same number of 1's. If not, append the word representing the shorter number to itself, i.e., double the shorter number. Repeat this procedure until both tapes get the same length. The desired least common multiple is on both tapes.

`TM-Input-Output`

Problem 18. We give the Turing machine $M = (Q, \Sigma, \Gamma, q_0, F, \delta)$ with $Q = \{q_0, q_1, q_2, q_3\}$, $\Sigma = \{0, 1\}$, $\Gamma = \{0, 1, \sqcup\}$, $F = \{q_3\}$ and the transition function

$$\delta \ : \ Q \times \Gamma \to Q \times \Gamma \times \{L, R\}$$

with $\delta(q_0, 1) = (q_0, 1, R)$, $\delta(q_0, 0) = \delta(q_1, 1) = (q_1, 1, R)$, $\delta(q_1, \sqcup) = (q_2, \sqcup, L)$, $\delta(q_2, 1) = (q_3, \sqcup, R)$. For any other values δ is not defined.
Write down the sequence of configurations of M starting on the word 110111.

Solution of Problem 18:

$q_0 110111 \vdash 1q_0 10111 \vdash 11q_0 0111 \vdash 111q_0 111 \vdash 1111q_1 11 \vdash 11111q_1 1 \vdash 11111q_1 1 \vdash 111111q_1 \sqcup \vdash 11111q_2 \sqcup\sqcup \vdash 11111 \sqcup q_3.$

`TM-2n-JSON`

Problem 19. Using the Turing Machine Simulator
`http://math.hws.edu/eck/js/turing-machine/TM.html`
construct a Turing machine M over $\Sigma = \{0\}$ which computes the "Double n" function $d : \mathbb{N} \to \mathbb{N}$ given by $d(n) = 2n$.
Use unary representation: A number n is represented by the string 0^n consisting of n copies of the symbol 0. As an extension of the Turing Machine concept, an infinite tape in both directions may be used. The solution shall be the JSON expression generated for your machine. For example, the JSON expression generated for the "Binary Increment" TM, copied from the above website is:

```
{
   "name": "Binary Increment",
   "max_state": 25,
   "symbols": "xyzabc01$@",
   "tape": "10011111",
   "position": 7,
   "rules": [
      [ 0, "#", "1", 1, "R" ],
      [ 0, "0", "1", 1, "R" ],
      [ 0, "1", "0", 0, "L" ],
      [ 1, "#", "#", "h", "L" ],
      [ 1, "0", "0", 1, "R" ],
      [ 1, "1", "1", 1, "R" ]
   ]
}
```

Solution of Problem 19:

```
{
   "name": "Double n",
   "max_state": 25,
   "symbols": "#0x",
   "tape": "0000",
   "position": 0,
   "rules": [
      [0, "0", "x", 2, "R"],
      [0, "#", "#", 1, "R"],
```

```
        [2, "0", "0", 4, "R"],
        [2, "#", "0", 3, "L"],
        [3, "x", "0", 1, "R"],
        [4, "0", "0", 4, "R"],
        [4, "#", "#", 5, "R"],
        [5, "0", "0", 5, "R"],
        [5, "#", "0", 6, "R"],
        [6, "0", "0", 6, "L"],
        [6, "x", "0", 7, "R"],
        [6, "#", "#", 6, "L"],
        [7, "0", "x", 2, "R"]
    ]
}
```

TMGenerate2n

Problem 20. Write down explicitly an enumerator G such that $\mathrm{Gen}(G) = \left\{0^{2n} \,\middle|\, n \in \mathbb{N}\right\}$.
Since in the lecture notes it has not been *formally* defined, how a Turing machine with two tapes works, you may describe the transition function as

$$\delta : Q \times \Gamma \to Q \times \Gamma \times \{R, L\} \times (\Gamma \cup \{\boxtimes\})$$

in the following way: If G is in state q and reads the symbol c from the working tape, and

$$\delta(q, c) = (q', c', d, c'')$$

then G goes to state q', replaces c by c' on the working tape and moves the working tape head in direction d. Moreover, unless $c'' = \boxtimes$, the symbol c'' is written on the output tape and the output tape head is moved one position forward. If, however, $c'' = \boxtimes$, nothing is written on the output tape and the output tape head rests in place.
Hint: There exists a solution with only 4 states.

Solution of Problem 20:

$G = (Q, \Gamma, \sqcup, \emptyset, \delta, q_0, \emptyset)$, $\Gamma = \{0, \sqcup, \#\}$, $Q = \{q_0, \ldots, q_3\}$.

Idea:

1. Write ## to the working tape.
2. Go back to the first #. (TMGenerate2n:repeat)
3. Copy every character until the last # to the output tape.
4. Go one position back and override # by 00#.
5. Go to 2. (TMGenerate2n:repeat)

M	0	#	$\sqcup$	
q_0			$q_1\#R\#$	Write first #.
q_1			$q_2\#L\boxtimes$	Write second #.
q_2	$q_2 0 L\boxtimes$	$q_3\#R\boxtimes$		Go back to first #.
q_3	$q_3 0 R 0$	$q_3 0 R\#$	$q_1 0 R\boxtimes$	Copy to output, then add two more 0's.

Other Solution:

Idea:

1. Write # to the working tape and to the output tape and go right. On the working tape that represents the beginning of the tape and on the output tape we have covered the empty word.
2. If on a blank, write down a 0 on the tape and a 0 on the output tape and go right. Turn into "left"-mode.
3. While in "left"-mode, copy every 0 you hit to the output tape and turn to "right"-mode at the initial #, which is also copied to the output tape.
4. In right mode, copy every 0 to the output tape, add another 0 to the end of the tape (also copied to the output tape) and go into "left"-mode.
5. etc.

M	0	$\#$	$\sqcup$	
q_0			$q_R\#R\#$	Write first #
q_R	q_R0R0		q_T0R0	add 0
q_T			$q_L \sqcup L\boxtimes$	turn to left-mode
q_L	q_L0L0	$q_R\#R\#$		while copying Go back to first #.

Of course, the states q_0 and q_L can be merged into one state.